EMBEDDABLE FLASH MEMORY SYSTEM FOR NON-VOLATILE STORAGE OF CODE, DATA AND BIT-STREAMS FOR EMBEDDED FPGA CONFIGURATIONS

BACKGROUND OF THE INVENTION

Field of the Invention

5   The present invention relates to an embeddable Flash memory system for non-volatile storage of code, data, and bit-streams for embedded FPGA configurations. More specifically, the invention relates to a memory system integrated into a single chip together with a microprocessor and including a modular array structure comprising a plurality of memory blocks.

Description of the Related Art

10   As is well known in this specific technical field, the continuous size and price reduction in hand-held digital equipment together with demanding computing performance and low power constraint for consumer applications is increasing the need for a technology that combines high performance digital

15   CMOS transistor and non-volatile flash memory.

For instance, an efficient power block for a memory device is disclosed in the article by R. Pelliconi, D. Iezzi, A. Baroni, M. Pasotti, P. L. Rolandi, "Power efficient charge pump in deep sub micron standard CMOS technology," Proceedings of 27th ESSCIRC, pp100-103, Sept. 2001.

20   At the same time raising costs of mask sets and shorter time-to-market available for new products are leading to the introduction of systems with a higher degree of programmability and configurability, such as system-on-chip with configurable processors, embedded FPGA, and embedded flash memory.

In this respect, the availability of an advanced embedded flash

25   technology, based on NOR architecture, together with innovative IP's, like embedded flash macrocells with special features, is a key factor.

For a better understanding of the present invention reference is made to the Field Programmable Gate Array (FPGA) technology combining standard processors with embedded FPGA devices.

These solutions enable configuration of the FPGA at deployment time with exactly the required peripherals, exploiting temporal re-use by dynamically reconfiguring the instruction-set at run time based on the currently executed algorithm.

The existing models for designing FPGA/processor interaction can be grouped in two main categories:

- the FPGA is a co-processor communicating with the main processor through a system bus or a specific I/O channel;

- the FPGA is described as a functional unit of the processor pipeline.

The first group includes the GARP processor, known from the article by T. Callahan, J. Hauser, and J. Wawrzynek entitled: "The Garp architecture and C compiler" IEEE Computer, 33(4) : 62–69, April 2000. A similar architecture is provided by the A-EPIC processor that is disclosed in the article by S. Palem and S. Talla entitled: "Adaptive explicit parallel instruction computing", Proceedings of the fourth Australasian Computer Architecture Conference (ACOAC), January 2001.

In both cases the FPGA is addressed via dedicated instructions, moving data explicitly to and from the processor. Control hardware is kept to a minimum, since no interlocks are needed to avoid hazards, but a significant overhead in clock cycles is required to implement communication.

Only when the number of cycles per execution of the FPGA is relatively high may the communication overhead be considered negligible.

In the commercial world, FPGA suppliers such as Altera Corporation offer digital architectures based on U.S. Patent No. 5,968,161, issued to T.J. Southgate, entitled: "FPGA based configurable CPU additionally including second programmable section for implementation of custom hardware support".

2

Other suppliers (Xilinx, Triscend) offer chips containing a processor embedded on the same silicon IC with embedded FPGA logic. See for instance U.S. Patent No. 6,467,009, issued to S.P. Winegarden et al., entitled: "Configurable Processor System Unit", and assigned to Triscend Corporation.

5 However, those chips are generally loosely coupled by a high speed dedicated bus, performing as two separate execution units rather than being merged in a single architectural entity. In this manner the FPGA does not have direct access to the processor memory subsystem, which is one of the strengths of the academic approaches outlined above.

10 In the second category (FPGA as a function unit) we find architectures commercially known as: "PRISC"; "Chimaera" and "ConCISe".

In all these models, data are read and written directly on the processor register file, minimizing overhead due to communication. In most cases, to minimize control logic and hazard handling and to fit in the processor pipeline

15 stages, the FPGA is limited to combinatorial logic only, thus severely limiting the performance boost that can be achieved.

These solutions represent a significant step toward a low-overhead interface between the two entities. Nevertheless, due to the granularity of FPGA operations and its hardware-oriented structure, their approach is still very coarse-

20 grained, reducing the possible resource usage parallelism and again including hardware issues not familiar nor friendly to software compilation tools and algorithm developers.

Thus, a relevant drawback in this approach is often the memory data access bottleneck that often forces long stalls on the FPGA device in order to fetch

25 on the shared registers enough data to justify its activation.

BRIEF SUMMARY OF THE INVENTION

The invention overcomes the limitations of similar preceding architectures by relying on a embedded device of a novel nature and a new design for processor/memory interface.

5   According to a first embodiment of the present invention, the embeddable Flash memory system includes a modular array structure having a plurality of memory blocks, wherein a power block, including charge pumps, is shared among different flash memory modules through a PMA arbiter in a multi-bank fashion.

10   Moreover, the embeddable Flash memory system according to the invention includes three different access ports, each for a specific function:

- a code port CP (10) optimized for random access time and the application system;

- a data port DP (11) allowing an easy way to access and modify

15   application data; and,

- an FPGA port FP (12) offering a serial access for fast download of bit streams for an embedded FPGA (e-FPGA) configurations.

In accordance with another embodiment of the invention, a memory system is provided that includes: a memory module, the memory module including

20   a non-volatile memory block, a power block, and a power management arbiter coupling the power block to the non-volatile memory block, the arbiter configured to provide charge pump sharing among flash memory modules in the non-volatile memory block from a single charge pump in the power block.

In accordance with another aspect of the present invention, a

25   memory system is provided that includes a plurality of non-volatile flash memory blocks; a power block comprising a plurality of charge pumps; a power management arbiter coupled to the power block and to the non-volatile flash memory blocks, the power management arbiter including a programming circuit configured to couple the charge pumps to the memory blocks and an order block

4

configured to order requests for the memory blocks; a crossbar coupled to the memory blocks, power block, and power management arbiter; a code port coupled to the crossbar and configured to optimize random access time and to manage application code stored in the memory blocks; a data port coupled to the crossbar

5   and configured to provide access to and modify application data stored in the memory blocks; a floating programmable gate array port coupled to the crossbar and configured to provide serial access for download of bit streams of embedded floating programmable gate array configurations stored in the memory modules; and a microprocessor coupled to the crossbar and configured to provide data

10   management and memory system testing.

In accordance with another aspect of the foregoing embodiment, the memory system further includes the order block being configured to order requests for the memory blocks in accordance with the following rules: status of the request (already active or new request); and priority information; and it preferably includes

15   the power management arbiter having a switch block coupled to an output of the order block and configured to manage requests of the memory blocks in response to output of the order block; and a request decoder block configured to enable the charge pumps; and a pump driver block configured to manage power down/stand-by timeout and to limit requests for each charge pump to a maximum allowed

20   number of requests.

Thus, the disclosed embodiments of the present invention provide a new embeddable memory architecture having functional and structural features capable of achieving significant performance and energy consumption enhancements with respect to a traditional signal processing device.

25

BRIEF DESCRIPTION OF THE DRAWINGS

The features and advantages of the digital architecture according to this invention will become apparent from the following description of a best mode

5

for carrying out the invention given by way of non-limiting example with reference to the enclosed drawings, wherein:

Figure 1 is a block diagram of a memory architecture for data storage processing according to the present invention;

5   Figure 2 is a block diagram of a programming circuit with a gate ramp slope dependent on current required by memory cells under programming;

Figure 3 is a schematic diagram of programming gate voltage ramp slope with 128 cells in programming (0.3 V/us);

Figure 4 is the real diagram of the voltage ramp slope of Figure 3;

10   Figure 5 is a block diagram of a sense amplifier;

Figure 6 is a block diagram of a power management block architecture; and

Figure 7 is a representation of a photograph of the memory architecture according to the present invention.

15   DETAILED DESCRIPTION OF THE INVENTION

With reference to the drawings views, generally shown in Figure 1 is an embeddable Flash memory system 1 for non-volatile storage of code, data and bit-streams for embedded FPGA configurations realized according to one embodiment of the present invention.

20   More specifically, an 8Mb application-specific embeddable flash memory system is disclosed. The memory system may be integrated into a single chip together with a microprocessor 2.

This memory architecture 1 includes three different access ports, each for a specific function:

25   - a code port 10 (CP), which is optimized for random access time and the application system; this port 10 may be used as in a usual flash memory;

- a data port 11 (DP) allowing an easy way to access and modify application data; and,

6

- an FPGA port 12 (FP) offering a serial access for a fast download of bit streams for embedded FPGA (e-FPGA) configurations.

A test chip will be further presented that is integrated for performance assessment as well as for design and built-in self-test methodology validation. A special automatic programming gate voltage ramp generator circuit that allows a programming rate of 1Mbyte/s and an erase time of 200ms is also introduced, as will be further clarified.

The memory system architecture 1 is schematically shown in Figure 1. The architecture comprises a modular memory 13 (dotted line) including charge pumps 14 (Power Block), testability circuits 16 (DFT), a power management arbiter 15 (PMA) and a customizable array of N independent 2Mb flash memory modules 3.

Depending on the storage requirements and performances, the number of modules 3 can be varied. In the current non-limiting example the number of modules has been chosen as N=4.

The modular memory 13 includes (N+2) 128-bit target ports and implements a N-bank uniform memory.

As previously mentioned, three content-specific ports 10, 11 and 12 are dedicated to code (CP, 64-bit wide), data (DP, 64-bit), and FPGA bit stream configurations (FP, 32-bit). A 128 bit sub-system crossbar 5 connects all the architecture blocks and the eight bit microprocessor 2.

The main features of such a flash memory system are: charge pump 14 sharing among different flash memory modules 3 through the PMA arbiter 15 in a multi-bank fashion. Moreover, the use of a small eight-bit micro processor 2 provides easy memory system testing and adds complex functionalities for data management, and the use of an ADC (Analog-to-Digital Converter), required by the application, increases the self-test capability of the system.

Below is an evaluation in more detail of the main features of the inventive memory architecture.

7

Each flash memory module 3 has a size of 2Mb and has a 128-bit IO data bus with 40ns access time, resulting in 400Mbyte/s overall throughput, and a program/erase control unit.

All the high-voltage generation section is in the power block 14,
5   which is shared by each of the four 2Mb flash memory modules 3.

A 1Mbyte/s programming rate with 128bit word requires that the programming charge pumps of the block 14 be able to supply up to 3mA of programming current.

These charge pumps are usually sized to sustain operations in worst
10   case conditions of process and temperature variations with all bits of a word in programming. This leads to an increase in the charge pump area of more than 130% with respect to typical conditions, when just half of the bits in a word will be programmed.

This memory architecture further includes a programming circuit 9
15   that overcomes this problem and is shown in Figure 2.

As may be appreciated, the memory cells 6 are organized in a memory matrix with associated row and column decoders. A multiplexer 7, fed by the output of a voltage regulator 17, will bias the memory matrix rows with a Vread voltage, while a program switch 8 fed by another voltage regulator 18 will bias the
20   memory matrix columns with a Vd voltage generated through a Vpd voltage value supplied by the charge pumps.

Referring now to Figures 3 and 4, after a preliminary program verify operation, phase A, the programming operation starts in phase B. The programming circuit 9 will move the gate voltage with a maximum slope defined by
25   the operational amplifier slew-rate, until the memory cells sink all the available current.

In a third phase C, the gate voltage reaches a level that switches on all the cells in programming and the charge pump output voltage Vpd lowers from 6V to 5V.

8

The operational amplifier changes the word-line voltage slope to fix the voltage Vpd at that voltage value where the charge pump can deliver current at maximum efficiency.

This current programs the flash cells 6 through the voltage regulator

5   18, which keeps the flash cell drain voltage Vd at a fixed value of 4.5V.

In phase C, all the cells 6 under programming will see their thresholds moving at the same rate and the generated programming gate voltage becomes a linear ramp, at first order, with an optimum slope defined by the current from the charge pump and by the number of bits in programming, enabling the

10   memory cells 6 to use all the available current, while modulating the programming gate voltage and, consequently, the programming speed.

As application data are supposed to be frequently modified, erased, and programmed, threshold distributions have been carefully positioned taking into account this assumption together with reliability and power consumption

15   considerations.

When a memory cell 6 is programmed or erased, power is consumed to move its threshold from the erased state to the programmed state, or from the programmed state to the erased state. The higher the voltage separating the two states, the more power is consumed to change the state of the cell.

20   With erased and programmed states voltage distance of about 2V and using an accurate sense amplifier, good reliability and power consumption performances can be obtained because programming and erasing algorithms are rapidly converging.

In typical conditions, when 64 bits are programmed out of 128 bits, a

25   programming time of less then 16ns is obtained. Figure 4 is a real diagram showing the worst case program operation of when all 128 bits of a word are programmed. As may be appreciated, it is completed in just one programming pulse even when the gate programming voltage reaches a relatively low voltage (~6V) and a total programming time of ~18ns for 16 bytes is obtained.

9

The erasing function takes full advantage from the programming circuit accelerating the soft-programming phase, and using large parallelism it is possible to have very short verify phases. A sector is erased in typically 200ms.

The sense amplifier shown in Figure 5 is of a known type and is able to operate down to 1.5V. This closed loop circuit enhances precision and current/voltage gain as needed to work with closer thresholds margins. A 40ns access time is obtained, or 400 Mbyte/s read rate, which allows a 32bit processor to run at up to 100 MHz.

The memory system 1 includes four 2 Mbit flash memory modules 3 that can be requested to perform one of three operations (read, write, erase) at the same time and independently. Simultaneous memory operations use the power management arbiter block 15 (PMA) for optimal scheduling.

Available power and user-defined priorities are considered to schedule conflicting resource requests in a single clock cycle.

Remembering that the write operation is composed by two different basic operations, program pulse and verify (a sequence that can be repeated), while the erase operation is composed by three different basic operations, erase pulse, verify (erase verify and depletion verify) and soft-program, each time a flash memory module enters a new basic operation, it sends a request to the PMA arbiter block 15 for all the needed high voltages allocation.

Read and verify are the only operations allowed to occur at the same time in the four flash memory modules 3, while the basic program pulse and basic erase pulse operations can be performed in just one memory module 3 at a time.

The main component of the PMA arbiter block 15, shown in Figure 6, is the order block 19. It orders the requests for the memory modules 3, following these rules:

status of the request (already active or new request);
priority information.

10

The requests are collected and processed in parallel by three stages: encoders, comparators, and one level of logic. The response is available in only one clock cycle.

Referring to Figure 6, a switch block 21 satisfies the requests of the flash memory modules 3 based on the order block 19 output. A request decoder is provided, block 20 req_dec, for enabling the required high voltage resources (charge pumps 14) while a corresponding pump driver block 22 manages the power down/stand-by timeout and limits the requests for each resource to the maximum allowed.

The correspondence between requests and high voltage resources, power down and standby time, and the maximum number of parallel requests that is possible to satisfy are configurable.

The function of the three ports 10, 11 and 12 will be described in detail.

The first port 10 is dedicated to manage application code stored in flash memory modules. It has also the possibility to write in the memory areas for .DP, to perform memory formatting, and FP, to store downloading configurations.

The code port CP 10 has four configuration registers defining its addressable memory space: two at the application level, and two at the flash memory modules level.

The I/O data word bus is 64 bits wide, while the address bus is 32 bits.

The port uses one chip select to access in the addressable memory space. During operations (read and write), the port acts as a conventional RAM memory, using a write enable in case of write operation. As this port allows the erase operation, which is necessary before a write operation in a flash memory modules, an erase enable input signal has been added.

During a read operation, an output ready signal is tied low when data are not available immediately, so that it can acts as a wait state signal.

11

The second data port DP 11 is dedicated to manage application data stored in flash memory modules, eventually organized in a file system by the application, using a typical data page of 512B.

The DP has four configuration registers defining its addressable

5   memory space: two at the application level, and two at the flash memory modules level.

The I/O data word bus is 64 bits wide, while the address bus is 32 bits.

Functions are offered to give the application the possibility to

10   implement a file system for data management.

The operations available are Read Page, Read Word, Write Page, Invalid Page and Defragmentation. A 512B SRAM page buffer allows the application to exchange data in burst mode at maximum speed to increase performance especially during write operation.

15   The erase operation is not available because it is hidden by the micro controller 2 that does a logic remapping of physical address.

Furthermore in Write Page, the physical address is chosen using an algorithm that takes into account the filling status of sectors. If there are full sectors with an invalid page, a defragmentation operation is automatically started

20   to increase free space, and sectors are eventually erased during this operation.

A Port Status Register is available and can be directly read in order to get information about the status of current operations.

The third FP port 12 is dedicated to manage embedded-FPGA (e-FPGA) configurations data stored in flash memory modules. The FP port is read-

25   only and provides fast sequential access for bit streams downloading.

The FP has four configuration registers replicating the information stored in the CP port 11 that must be used in order to write e-FPGA configurations data.

The output data word bus and the address bus are 32 bits wide. The FP port 12 uses a chip select to access in the addressable memory space, and a burst enable to allow burst serial access.

In read operation, an output ready signal is tied low when data are
5   not immediately available, so that it can act as a wait state signal.

The eight-bit microprocessor 2 (uP) performs additional complex functions (defragmentation, compression, virtual erase, etc.) not natively supported by the DP port 11, and assists in a built-in self test of the memory system. The (N+2)x4 128-bit crossbar 15 connects the modular memory with the four initiators
10   (CP, DP, FP and uP) providing that the three and preferably four flash memory modules 3 can be read in parallel at full speed.

The memory space of the four flash memory modules 3 is arranged in three programmable user-defined partitions, each one devoted to a port. The memory system clock can run up to 100 MHz, and reading three modules 3 with
15   128 bit data bus and 40 ns access time, results in a peak read throughput of 1.2 GB/s.

The overall system testability is enhanced by the specific DFT block 16 connected to all relevant internal signals. It makes use of an external high voltage power supply, while access from the external test equipment is granted by
20   two analog IO pads (not shown).

By means of external analog references, the DFT block 16 can first test its own circuitry and then all internally generated voltages and currents that are vital for the correct system operations (*e.g.*, band gap voltage, regulated voltages, charge pumps).
25   The measurement capability of the component can be profitably applied to the trimming of analog internal signals, so that also the following operations can be implemented:

- reference flash cell current measurement and calibration;
- voltage and current reference calibration;

13

- threshold voltage (as obtained by an analog sense amplifier) measurement of memory cells.

See, for example, the article by P. L. Rolandi et al., "1M-cell 6b/cell analog flash memory for digital storage", ISSCC 1998 Digest of Technical Papers,

5   pp 334-335, Feb. 1998.

The test flux is controlled by the microprocessor 2 present in the system.

The main components of the DFT block 16 are a network of analog switches, multiplexers, a charge integrator, a voltage attenuator, a comparator and

10   a ten bit pipeline ADC (Analog-to-Digital Converter).

The two analog IO pads have the main function to provide external references for the measurements. But they also allow, by means of the analog switches network, a wide direct detectability of internal nodes under test in the system.

15   The voltage path to the ADC is fully differential, yielding advantages in terms of power supply noise rejection.

Hereinafter, Table I resumes the technology parameters and device performance of the inventive memory architecture, while in Figure 7 it is shown a picture of the test chip that has been designed using a NOR type 0.18μm flash

20   embedded technology with 1.8V power supply, two poly, six metal and memory cell size of $0.35\mu m^2$. The test chip size is 8.4x4.8 mm^2.

TABLE I

TECHNOLOGY AND DEVICE PARAMETERS

Process	0.18μm CMOS, two poly, six metal
Tunneling oxide	10nm
Cell size	0.35μm^2
Organization	Four modules x 256Kb x nine sectors
Memory module word	128 bits
Supply voltage	1.6V- 2.0V
Program throughput	1MB/s
Sector erasing time	200ms
Access time	40ns
Peak read throughput	1.2GB/s

5   From the previous description it may be appreciated that the memory architecture has a whole size of 8Mb application-specific embeddable flash memory cells and comprises three content-specific I/O ports that can deliver a peak read throughput of 1.2GB/s for non-volatile storage of code, data and embedded FPGA bit stream configurations.

10   All of the above U.S. patents, U.S. patent application publications, U.S. patent applications, foreign patents, foreign patent applications and non-patent publications referred to in this specification and/or listed in the Application Data Sheet, are incorporated herein by reference, in their entirety.

        From the foregoing it will be appreciated that, although specific

15 embodiments of the invention have been described herein for purposes of illustration, various modifications may be made without deviating from the spirit and scope of the invention. Accordingly, the invention is not limited except as by the appended claims.